

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the Application of:

JAE-BUM KO, ET AL.

Application No.:

Filed:

For: **Semiconductor Memory Device  
Having Tag Block for Reducing  
Initialization Time**

Art Group:

Examiner:

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**REQUEST FOR PRIORITY**

Sir:

Applicant respectfully requests a convention priority for the above-captioned application, namely:

COUNTRY	APPLICATION NUMBER	DATE OF FILING
Republic of Korea	2003-27653	30 April 2003

A certified copy of the document is being submitted herewith.

Respectfully submitted,

Blakely, Sokoloff, Taylor & Zafman LLP

Dated: 12/30/03

12400 Wilshire Boulevard, 7th Floor  
Los Angeles, CA 90025  
Telephone: (310) 207-3800

  
William Thomas Babbitt, Reg. No. 39,591



별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto  
is a true copy from the records of the Korean Intellectual  
Property Office.

출 원 번 호 : 10-2003-0027653  
Application Number

출 원 년 월 일 : 2003년 04월 30일  
Date of Application APR 30, 2003

출 원 인 : 주식회사 하이닉스반도체  
Applicant(s) Hynix Semiconductor Inc.



2003 년 10 월 24 일

특 허 청  
COMMISSIONER

## 【서지사항】

【서류명】	특허출원서
【권리구분】	특허
【수신처】	특허청장
【참조번호】	0007
【제출일자】	2003.04.30
【발명의 명칭】	초기화 동작시간이 감소된 태그블록을 구비하는 반도체 메모리 장치
【발명의 영문명칭】	SEMICONDUCTOR MEMORY DEVICE WITH TAG BLOCK FOR REDUCING TIME OF INITIALIZATION OPERATION
【출원인】	
【명칭】	주식회사 하이닉스반도체
【출원인코드】	1-1998-004569-8
【대리인】	
【명칭】	특허법인 신성
【대리인코드】	9-2000-100004-8
【지정된변리사】	변리사 정지원, 변리사 원석희, 변리사 박해천
【포괄위임등록번호】	2000-049307-2
【발명자】	
【성명의 국문표기】	고재범
【성명의 영문표기】	KO, Jae Bum
【주민등록번호】	760926-1026025
【우편번호】	467-862
【주소】	경기도 이천시 부발읍 신하6리 삼익세라믹아파트 102-208
【국적】	KR
【발명자】	
【성명의 국문표기】	안진홍
【성명의 영문표기】	AWN, Jin Hong
【주민등록번호】	581124-1110419
【우편번호】	431-070
【주소】	경기도 안양시 동안구 평촌동 130-1 영풍아파트 101-1408
【국적】	KR

## 【발명자】

【성명의 국문표기】 홍상훈  
 【성명의 영문표기】 HONG, Sang Hoon  
 【주민등록번호】 700930-1064113  
 【우편번호】 467-860  
 【주소】 경기도 이천시 부발읍 신하리 청구 아파트 101-1302  
 【국적】 KR

## 【발명자】

【성명의 국문표기】 김세준  
 【성명의 영문표기】 KIM, Se Jun  
 【주민등록번호】 740209-1069419  
 【우편번호】 463-500  
 【주소】 경기도 성남시 분당구 구미동 까치마을 선경아파트 107-1002  
 【국적】 KR  
 【심사청구】 청구  
 【취지】 특허법 제42조의 규정에 의한 출원, 특허법 제60조의 규정에 의한 출원심사 를 청구합니다. 대리인  
 특허법인 신성 (인)

## 【수수료】

【기본출원료】	20	면	29,000	원
【가산출원료】	7	면	7,000	원
【우선권주장료】	0	건	0	원
【심사청구료】	3	항	205,000	원
【합계】	241,000			원
【첨부서류】	1. 요약서·명세서(도면)_1통			

### 【요약서】

#### 【요약】

본 발명은 태그메모리를 구비하여 고속 동작이 가능한 메모리 장치에서 태그메모리의 초기화시간을 줄일 수 있는 메모리 장치를 제공하기 위한 것으로, 이를 위해 본 발명은 각각 M개의 워드라인을 가지며, 입력되는 로우어드레스에 대응하도록 구비된 N개의 단위셀블럭에, 추가적으로 M개의 워드라인을 가지기 위해 추가의 단위셀블럭을 더 포함하여 N+1개의 단위셀블럭으로 구성되는 셀블럭; 상기 (N+1) 개의 워드라인 중에서, 적어도 어느 한 워드라인을 예비워드라인으로 할지의 여부에 대한 정보를 저장하기 위한 예비셀블럭 테이블; 상기 로우어드레스를 입력받아 N개의 단위셀블럭을 선택하기 위한 논리적 셀블럭어드레스를 감지하고, 이를 상기 N+1개의 단위셀블럭중 하나를 선택하기 위한 물리적 셀블럭어드레스로 변환하여 출력하는 태그블럭; 및 상기 물리적 셀블럭어드레스에 의해 선택된 단위셀블럭에서의 한 워드라인과, 상기 워드라인에 대한 -상기 예비셀블럭 테이블로부터 제공되는 정보에 의해 정해지는- 예비워드라인을 활성화시키기 위해 상기 태그블럭과 상기 예비셀블럭 테이블을 제어하는 제어수단을 구비하며, 상기 태그블럭은 상기 N+1개의 단위셀블럭에 각각 대응하여, 단위셀블럭에 각각 구비된 256개의 워드라인이 어떤 논리적 셀블럭어드레스에 대응되는지를 저장하기 위한 N+1개의 단위태그테이블; 및 한번의 초기화 동작으로 상기 N+1개의 단위태그테이블을 초기화시키기 위한 초기화 셋팅수단을 구비하는 메모리 장치를 제공한다.

#### 【대표도】

도 6

●

1020030027653

출력 일자: 2003/10/29

【색인어】

반도체, 메모리, 초기화, 예비워드라인, 태그메모리.

**【명세서】****【발명의 명칭】**

초기화 동작시간이 감소된 태그블럭을 구비하는 반도체 메모리 장치{SEMICONDUCTOR MEMORY DEVICE WITH TAG BLOCK FOR REDUCING TIME OF INITIALIZAION OPERATION}

**【도면의 간단한 설명】**

도1은 종래기술에 의한 메모리 장치의 블럭구성도.

도2는 도1에 도시된 태그블럭의 블럭구성도.

도3은 도1에 도시된 예비셀블럭 테이블의 블럭구성도.

도4는 도2에 도시된 단위태그테이블을 나타내는 블럭구성도.

도5는 도1에 도시된 메모리 장치의 동작을 나타내는 동작 흐름도.

도6는 본 발명의 바람직한 실시예에 따른 메모리 장치의 블럭구성도.

도7은 도6에 도시된 초기화세팅부를 나타내는 블럭구성도.

**【발명의 상세한 설명】****【발명의 목적】****【발명이 속하는 기술분야 및 그 분야의 종래기술】**

<8> 본 발명은 반도체 메모리 장치에 관한 것으로, 특히 태그메모리를 사용하여 고속으로 데이터 억세스가 가능한 반도체 장치의 초기화 동작에 관한 것이다.

<9> 근래에 중앙처리장치(CPU)의 동작속도는 메모리 장치(DRAM)의 동작속도를 능가할 정도로 현저히 향상되어 왔으며, 그 결과 메모리 장치의 동작속도가 중앙처리장치의 동작속도보다 상대적으로 느려 여러가지 문제점이 발생하고 있다. 이러한 문제점을 극복하기 위해 보다 고속으로 데이터를 입출력하기 위한 다양한 구조의 메모리 장치가 개발되고 있다.

<10> 도1은 본 출원인이 2003년 2월 21일에 출원한 고속으로 데이터 액세스가 가능한 메모리 장치(출원번호 2003-11121)의 블럭구성도이다.

<11> 도1을 참조하여 살펴보면, 고속으로 데이터 액세스가 가능한 메모리 장치는 각각 256개의 워드라인을 가지며, 입력되는 로우어드레스에 대응하도록 구비된 8개의 단위셀블럭에, 추가적으로 256개의 워드라인을 가지기 위해 추가의 단위셀블럭을 더 포함하여 9개의 단위셀블럭으로 구성되는 셀블럭(500)과, 셀블럭에(500) 구비되는 (8+1) × 256개의 워드라인 중에서, 적어도 어느 한 워드라인을 예비워드라인으로 할지의 여부에 대한 정보를 저장하기 위한 예비셀블럭테이블(410)과, 상기 로우어드레스를 입력받아 8개의 단위셀블럭을 선택하기 위한 논리적 셀블럭어드레스를 감지하고, 이를 9개의 단위셀블럭중 하나를 선택하기 위한 물리적 셀블럭어드레스로 변환하여 출력하기 위한 태그블럭(430)과, 물리적 셀블럭어드레스에 의해 선택된 단위셀블럭에서의 한 워드라인과, 상기 워드라인에 대한 -상기 예비셀블럭 테이블로부터 제공되는 정보에 의해 정해지는- 예비워드라인을 활성화시키기 위해 태그블럭(430)과 예비셀블럭 테이블(410)을 제어하는 제어부(420)를 구비한다.

<12> 여기서 제어부(420)는 9개의 단위셀블럭중에서 선택된 하나의 단위셀블럭에 연속적으로 제1 및 제2 데이터가 액세스될 때, 제1 데이터에 대한 재저장동작은 제1 데이터를 액세스하기 위해 활성화된 워드라인에 대응하여 지정된 예비워드라인을 통해 이루어지도록 제어하게 된다.

<13> 또한 도에 도시된 메모리 장치의 제어부(420)에 입력되는 어드레스는 8 × 256개의 워드라인에 대응하고, 추가의 단위셀블럭의 구비되는 워드라인의 수만큼인 256개의 워드라인을 예비워드라인으로 지정하게 된다. 예비워드라인은 한 단위셀블럭에 있는 256개의 워드라인으로 고정적으로 정해지는 것은 아니고 메모리 장치가 동작중에서 계속해서 위치가 변하게 되는데, 이에 대해서는 뒤에서 자세히 설명한다.

<14> 도2는 도1에 도시된 태그블럭의 블럭구성도이다.

<15> 도2를 참조하여 살펴보면, 태그블럭(430)은 9개의 단위셀블럭(510 ~ 590)에 각각 구비된 256개의 워드라인이 어떤 논리적 셀블럭에 대응되는 것인지를 각각 저장하고 있는 9개의 단위태그테이블(432a~432i)과, 입력되는 논리적셀블럭어드레스(Cur\_LBA)와, 입력되는 로컬어드레스(Cur\_RA)에 대응되어 9개의 단위태그테이블(432a~432i)에서 출력되는 정보 -로컬어드레스(Cur\_RA)에 의해 선택된 워드라인이 어떤 논리적셀블럭에 관한 것에 관한 9개의 데이터 정보-를 각각 비교하기 위한 9개의 비교부(433a~433i)와, 9개의 비교부(433a~433i)에서 비교한 정보를 인코딩하여 물리적 셀블럭어드레스(Cur\_PBA)를 출력하기 위한 인코더(434)와 9개의 단위태그테이블(432a~432i)과, 9개의 비교부(433a~433i)와 인코더(434)를 제어하기 위한 테그제어부(436)를 구비한다.

<16> 태그블럭(430)은 예비셀블럭 테이블(420)에서 출력되는 정보(Extra\_BA) -로컬어드레스(Cur\_RA)에 대응하는 워드라인의 예비워드라인이 9개의 단위셀블럭(500)중에서 어떤 단위셀블럭에 있는지에 대한 정보-를 디코딩하여, 9개의 단위태그테이블(432a~432i)중에서 하나의 단위태그테이블을 선택하기 위한 신호를 출력하는 셀블럭어드레스 디코더부(431)를 더 구비한다. 또한 태그블럭(430)은 현재 동작클럭(CK)의 물리적 셀블럭어드레스(Cur\_PBA)를 한클럭 지연시

켜 이전 클럭의 물리적 셀블럭어드레스(Pre\_PBA)를 출력하기 위한 태그지연부(435)를 구비한다.

<17> 여기서 각각의 단위태그테이블(432a~432i)은 256개의 레지스터를 구비하고, 논리적 셀블럭어드레스가 8개이므로 각각의 레지스터는 3비트로 구성된다. 예를 들어 제1 단위태그테이블(432a)은 제1 단위셀블럭에 구비된 256개의 워드라인이 어떤 논리적 셀블럭에 대한 것인지를 저장하고 있고, 제2 단위태그테이블(432b)는 제2 단위셀블럭에 구비된 256개의 워드라인이 각각 어떤 논리적 셀블럭에 대한 것인지를 저장하고 있는 것이다.

<18> 각각의 단위태그테이블(432a~432i)에서 첫번째 레지스터(0)는 제1 단위셀블럭의 워드라인 'WL0'의 논리적 셀블럭어드레스를 저장하고 두번째 레지스터(1)는 제1 단위셀블럭의 워드라인 'WL1'의 논리적 셀블럭어드레스를 저장하고, 제256번째 레지스터(255)는 제1 단위셀블럭의 워드라인 'WL255'의 논리적 셀블럭어드레스를 저장한다.

<19> 예컨대 도2에 도시된 제1 단위태그테이블(432a)을 참조하여 살펴보면, 첫번째 레지스터에 저장된 데이터 '0'으로 인하여 제1 단위셀블럭의 워드라인(WL0)은 제2 논리적 단위셀블럭의 워드라인(WL0)에 대응하는 데이터라는 뜻이고, 255번째 레지스터에 저장된 데이터 '7'로 인하여, 제1 단위셀블럭의 워드라인(WL255)은 제8 논리적 단위셀블럭의 워드라인(WL255)에 대응하는 데이터인 것이다.

<20> 도3은 도1에 도시된 예비셀블럭 테이블(410)의 블럭구성도이다.

<21> 도3을 참조하여 살펴보면, 예비셀블럭 테이블(410)은 256개의 예비워드라인이 9개의 물리적 단위셀블럭중에서 어떤 단위셀블럭에 있는지에 대한 정보를 저장하기 위한 256개의 레지스터(411)를 구비한다. 256개의 워드라인(WL)에 대한 예비워드라인 정보를 저장하기 위해 예비

셀블럭 테이블(410)에는 256개의 레지스터(411)가 있고, 각각의 레지스터(411)는 9개의 단위 셀블럭에 대한 물리적 셀블럭어드레스에 대한 정보를 저장해야 하기 때문에 각각 4비트로 구현된다.

<22> 예비 셀블럭 테이블(410)에 저장된 내용을 살펴보면, 워드라인(WL0)의 예비워드라인은 제2 단위 셀블럭의 워드라인 'WL0'이라는 것이고(첫번째 레지스터(0) 참조), 워드라인(WL3)의 예비워드라인은 제4 단위 셀블럭의 워드라인 'WL3'이라는 뜻이다(네번째 레지스터(3) 참조). 메모리 장치의 동작중에 레지스터(411)에 저장된 내용은 계속 업데이트되며, 업데이트 될 때마다 256개의 예비워드라인에 대한 정보는 계속 바뀌게 된다.

<23> 또한, 예비 셀블럭 테이블(410)는 현재 실행단계의 로컬어드레스(Cur\_RA)를 입력 받아 현재 실행되고 있는 로컬어드레스(Cur\_RA)에 대한 예비워드라인이 어떤 단위 셀블럭에 지정되었는지를 알려주는 예비블럭 신호(Extra\_BA)를 태그블럭(430)으로 출력한다. 또한, 예비 셀블럭 테이블(410)는 이전 실행단계의 로컬어드레스(Pre\_RA)와, 이전 실행단계의 물리적 셀블럭어드레스(Pre\_PBA)를 입력 받아 예비블럭 업데이트 신호(EBT\_UPDATE)에 응답하여 레지스터(411)를 업데이트하게 된다.

<24> 도4는 도2에 도시된 단위태그테이블 중 하나(432a)를 나타내는 블럭구성도이다.

<25> 도4를 참조하여 살펴보면, 단위태그테이블(432a)은 셀블럭어드레스 디코더부(431)에서 출력되는 디코딩된 신호(dec\_Extra\_BA)에 의해 인에이블되고, 논리적 셀블럭어드레스(Cur\_LBA)와, 로컬어드레스(Cur\_RA)를 입력 받는다. 로컬어드레스(RA)는 로우어드레스 중에서는 한 단위 셀블럭에 구비된 256개의 워드라인 중 하나를 선택하기 위한 어드레스이다.

<26> 하나의 단위 태그테이블(432a)은 각각 3비트의 데이터를 저장할 수 있는 256개의 레지스터를 구비하는데, 입력되는 로컬어드레스(RA)에 의해 256개의 레지스터중 하나가 선택되고, 선택된 레지스터에 저장된 데이터가 출력 논리적 셀블럭어드레스(LBA\_OUT)로 된다. 또한 레지스터에 구비된 데이터를 업데이터 시킬 때에는 입력된 로컬어드레스(RA)에 의해 선택된 레지스터에, 입력되는 논리적 셀블럭어드레스(Cur\_LBA)를 저장하게 된다.

<27> 도5는 도1에 도시된 메모리 장치의 동작을 나타내는 동작 흐름도이다. 이하 도1 내지 도5를 참조하여 도4에 도시된 메모리 장치의 동작에 대해서 설명한다.

<28> 도1에 도시된 메모리 장치는 입력되는 어드레스에 대응하는 단위 셀블럭이 8개인 경우에 하나의 단위 셀블럭을 더 구비하여 총 8+1개의 단위 셀블럭을 구비한다. 추가적으로 구비되는 하나의 단위 셀블럭은 동일한 단위 셀블럭에 연속해서 데이터 억세스가 일어날 때 이전 명령어에 의해 억세스된 데이터의 재저장을 위한 것이다.

<29> 도5에서는 8+1개의 단위 셀블럭 중에서 제1 단위 셀블럭에 연속적으로 데이터가 억세스되고, 입력되는 어드레스에 의해 워드라인 WL0, WL1이 연속해서 활성화되는 것으로 가정하였다.

<30> 먼저 제1 명령어(CD0)에 의해 8+1개의 단위 셀블럭 중에서 선택된 제1 단위 셀블럭의 제1 워드라인(WL0)이 활성화된다. 이어서, 제1 워드라인(WL0)에 대응되는 K개의 데이터를 제1 단위 셀블럭에 구비된 센스앰프로 감지 증폭한다. 이어서 제1 워드라인(WL0)에 대응하는 K개의 데이터를 데이터래치부(450)로 이동시켜 래치시킨다. 이어서, 제1 워드라인(WL0)에 대응하여 감지 증폭된 K개의 데이터가 래치된 제1 단위 셀블럭의 센스앰프를 강제프리차지시킨다.

<31> 이어서 제1 명령어(CD0)가 리드명령어인 경우에는 데이터래치부(20)에 래치된 K개의 데이터 중 선택된 하나의 데이터를 제1 명령어(CD0)에 대응하는 출력 데이터로 출력시키고, 제1 명

령어(CD0)가 라이트명령어인 경우에는 제1 명령어(CD0)에 대응하여 입력되는 데이터로 데이터래치부(20)에 래치된 K개의 데이터중 선택된 하나의 데이터를 덮어쓰게 된다.

<32> 이어서 제1 워드라인(WL0)에 대응하는 예비워드라인으로 지정된 제2 단위셀블럭의 워드라인(WL0)을 활성화시킨다. 이어서 데이터래치부(450)에 래치된 K개의 데이터를 제2 단위셀블럭의 워드라인(WL0)에 대응되는 K개의 단위셀에 재저장한다.

<33> 이어서 제2 명령어(CD1)에 대응하여 선택된 제1 단위셀블럭의 제2 워드라인(WL1)을 활성화시킨다. 이어서 제2 워드라인(WL1)에 대응하는 K개의 데이터를 감지증폭한다.

<34> 여기서 제1 명령어(CD0)에 대응하여 감지, 증폭된 K개의 데이터가 재저장 동작과 제2 명령어(CD1)에 대응하여 K개의 데이터가 감지, 증폭되는 동작은 실질적으로 같은 타이밍에 이루어진다.

<35> 이렇게 데이터를 억세스 함으로서 이전명령어에 의해 감지, 증폭된 K개의 데이터가 재저장되는 타이밍에 상관없이 다음 명령어에 의해 데이터를 감지, 증폭할 수 있게 되어, 데이터 재저장시간만큼 데이터억세스 시간을 줄일 수 있게 되는 것이다.

<36> 한편 입력되는 명령어에 의해 활성화되는 워드라인에 대해서는 하나의 예비워드라인이 필요하게 되는데, 이는 예비셀블럭 테이블에 저장되어 있다. 예비워드라인으로 지정되는 워드라인의 수는 하나의 단위셀블럭에 구비되는 워드라인의 수인 256개이다. 하나에 단위셀블럭 구비된 모든 워드라인이 예비워드라인이 고정되는 것이 아니고, 메모리 장치의 동작중에 계속 바뀌게 되는 것이다. 이는 어드레스에 대응하는 단위셀블럭보다 추가적으로 하나의 단위셀블럭을 구비함으로서 가능하게 된다.

<37> 이어서 제1 데이터 및 제2 데이터가 제1 및 제2 단위셀블럭에서 교대로 억세스되는 인터리빙모드로 동작하는 것에 대해서 살펴본다.

<38> 먼저 제1 데이터를 8+1개의 단위셀블럭중에서 선택된 제1 단위셀블럭에서 감지 증폭한 다음 데이터래치부(450)에 래치시킨다. 이어서 제1 단위셀블럭에 제1 데이터를 재저장시키고, 상기 재저장 타이밍과 실질적으로 동일한 타이밍에 상기 8+1개의 단위셀블럭중에서 선택된 제2 단위셀블럭에서 상기 제2 데이터를 감지 증폭하여 데이터래치부(450)에 래치시킨다.

<39> 따라서 메모리장치는 데이터가 하나의 셀에 계속해서 억세스되는 경우에는 이전 데이터에 대한 재저장은 지정된 예비워드라인에 대응하는 단위셀에서 수행하고, 데이터가 단위셀에 교대로 억세스되는 경우에는 인터리빙모드로 동작함으로서, 억세스되는 패턴에 상관없이 고속으로 데이터를 억세스 할 수 있게 되는 것이다.

<40> 도1에 도시된 메모리 장치는 입력되는 셀블럭어드레스에 대응하는 8개의 단위셀블럭에다 추가적으로 하나의 단위셀블럭을 더 구비하기 때문에, 입력되는 셀블럭어드레스는 논리적 셀블럭어드레스를 뜻하는 것이고, 이를 실제 9개의 물리적 단위셀블럭 중 하나를 선택하는 물리적 셀어드레스로 변환하는 변환부가 필요하게 되는데, 제어부(420)에서 태그블럭(430)을 제어하여 셀블럭어드레스 변환 동작을 하게 된다.

<41> 또한, 물리적 셀블럭어드레스를 저장하고 있는 저장부가 필요하게 되는데, 이 때의 저장부가 태그블럭(430)이며, 전술한 바와 같이 256개의 예비워드라인에 대한 정보를 저장하고 있는 저장부는 예비셀블럭테이블이다. 8+1개의 단위셀블럭이 있고, 하나의 단위셀블럭에는 256개의 워드라인이 구비된다면, 태그블럭(430)에는 9개의 단위태그테이블이 구비되고 하나의 단위태그테이블에는 256 × 8비트의 메모리셀이 필요하며, 예비블럭테이블에는 256 × 4비트의 메모리셀이 필요하다.

<42> 한편, 도1에 도시된 메모리 장치의 초기 동작시에 태그블럭에 구비된 단위태그테이블도 초기화를 해야한다. 각각 9개의 단위태그테이블(432a ~ 432i)에 구비되는 256개의 레지스터를 초기화하기 위해서는 먼저 하나의 단위태그테이블(도4의 432a 참조)을 선택해야 한다. 이어서 선택된 단위태그테이블(432a)에 구비되는 256개의 레지스터 중 하나를 선택하기 위해서 로컬어드레스 신호(Cur\_RA)를 입력하고, 초기화될 물리적 셀블럭어드레스(Cur\_LBA)를 입력하게 된다.

<43> 예를 들어 로컬어드레스 신호(Cur\_RA)를 0 ~ 255 까지 순차적으로 증가시키면서 입력시켜 256개의 레지스터를 순차적으로 선택하고, 초기화될 물리적 셀블럭어드레스(Cur\_LBA)로는 '0'을 계속해서 입력시키는 것이다. 단위태그테이블(432a)의 초기화가 끝나면, 단위태그테이블(432b)도 같은 방법으로 초기화를 한다.

<44> 따라서 9개의 단위태그테이블을 초기화하려면 모두 256 × 9번의 초기화 동작이 필요한 것이다. 고속으로 동작하기 위해 태그메모리를 구비하는 메모리 장치에서 오히려 태그메모리의 초기화 동작이 고속동작을 방해하게 되는 문제점이 생긴다.

### 【발명이 이루고자 하는 기술적 과제】

<45> 본 발명은 태그메모리를 구비하여 고속 동작이 가능한 메모리 장치에서 태그메모리의 초기화시간을 줄일 수 있는 메모리 장치를 제공함을 목적으로 한다.

### 【발명의 구성 및 작용】

<46> 본 발명은 상기의 과제를 해결하기 위해 각각 M개의 워드라인을 가지며, 입력되는 로우어드레스에 대응하도록 구비된 N개의 단위셀블럭에, 추가적으로 M개의 워드라인을 가지기 위해

추가의 단위셀블럭을 더 포함하여 N+1개의 단위셀블럭으로 구성되는 셀블럭; 상기 (N+1) 개의 워드라인 중에서, 적어도 어느 한 워드라인을 예비워드라인으로 할지의 여부에 대한 정보를 저장하기 위한 예비셀블럭 테이블; 상기 로우어드레스를 입력받아 N개의 단위셀블럭을 선택하기 위한 논리적 셀블럭어드레스를 감지하고, 이를 상기 N+1개의 단위셀블럭중 하나를 선택하기 위한 물리적 셀블럭어드레스로 변환하여 출력하는 태그블럭; 및 상기 물리적 셀블럭어드레스에 의해 선택된 단위셀블럭에서의 한 워드라인과, 상기 워드라인에 대한 -상기 예비셀블럭 테이블로부터 제공되는 정보에 의해 정해지는- 예비워드라인을 활성화시키기 위해 상기 태그블럭과 상기 예비셀블럭 테이블을 제어하는 제어수단을 구비하며, 상기 태그블럭은 상기 N+1개의 단위셀블럭에 각각 대응하며, 단위셀블럭에 각각 구비된 256개의 워드라인이 어떤 논리적 셀블럭어드레스에 대응되는지를 저장하기 위한 N+1개의 단위태그테이블; 및 한번의 초기화 동작으로 상기 N+1개의 단위태그테이블을 초기화시키기 위한 초기화 셋팅수단을 구비하는 메모리 장치를 제공한다.

<47> 이하, 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자가 본 발명의 기술적 사상을 용이하게 실시 할 수 있을 정도로 상세히 설명하기 위하여, 본 발명의 가장 바람직한 실시 예를 첨부된 도면을 참조하여 설명하기로 한다.

<48> 도6는 본 발명의 바람직한 실시예에 따른 메모리 장치의 블럭구성도이다.

<49> 본 실시예에 따른 메모리 장치는 각각 256개의 워드라인을 가지며, 입력되는 로우어드레스에 대응하도록 구비된 8개의 단위셀블럭에, 추가적으로 256개의 워드라인을 가지기 위해 추가의 단위셀블럭을 더 포함하여 8+1 개의 단위셀블럭으로 구성되는 셀블럭(500)과, (8+1) × 256 개의 워드라인 중에서, 적어도 어느 한 워드라인을 예비워드라인으로 할지의 여부에 대한 정보

를 저장하기 위한 예비셀블럭 테이블(410)과, 로우어드레스를 입력받아 8개의 단위셀블럭중 하나를 선택하기 위한 논리적 셀블럭어드레스를 감지하고, 이를 8+1개의 단위셀블럭(500)중 하나를 선택하기 위한 물리적 셀블럭어드레스로 변환하여 출력하는 태그블럭(460)과, 물리적 셀블럭어드레스에 의해 선택된 단위셀블럭에서의 한 워드라인과, 워드라인에 대한 - 예비셀블럭 테이블(410)로부터 제공되는 정보에 의해 정해지는- 예비워드라인을 활성화시키기 위해 태그블럭(460)과 예비셀블럭 테이블(410)을 제어하는 제어부(420)를 구비한다.

<50> 태그블럭(460)은 8+1개의 단위셀블럭에 각각 대응하며, 단위셀블럭에 각각 구비된 256개의 워드라인이 어떤 논리적 셀블럭어드레스에 대응되는지를 저장하기 위한 8+1개의 단위태그테이블(432a ~ 432i)과, 한번의 초기화 동작으로 8+1개의 단위태그테이블(432a ~ 432i)을 초기화시키기 위한 초기화 셋팅부(461)을 구비한다.

<51> 도7은 도6에 도시된 초기화세팅부를 나타내는 블럭구성도이다.

<52> 도7을 참조하여 살펴보면, 초기화 셋팅부(461)는 8+1개의 단위태그테이블(432a ~ 432i)에 대응하며, 일측 입력으로 8+1개의 단위태그테이블(432a ~ 432i) 모두를 선택하기 위한 초기화 선택신호(IE)를 입력받고, 타입력으로 8+1개의 단위태그테이블(432a ~ 432i)중 하나를 선택하기 위한 선택신호(dec\_Extra\_BA0, dec\_Extra\_BA1, ...)를 각각 입력받는 다수의 논리합게이트(461\_1a, 461\_2a, ...)와 8+1개의 단위태그테이블(432a ~ 432i)에 대응하며, 초기화 선택신호(IE)에 응답하여 논리적 셀블럭어드레스(Cur\_LBA)와 8+1개의 단위태그테이블(432a ~ 432i)을 초기화시키기 위한 초기화 신호(BN0, BN1, ...)를 선택적으로 단위태그테이블(432a ~ 432i)에 출력하는 다수의 제1 멀티플렉서(461\_1b, 461\_2b, ...)와, N+1개의 단위태그테이블(432a ~ 432i)에 대응하여, 초기화 선택신호(IE)에 응답하여 단위셀블럭(432a ~ 432i)에 구비되는 256개의 워드라인을 선택하기 위한 로컬어드레스(Cur\_RA)와, 하나의 단위태그테이블에 구비되는 모든 레지스

터(도2 참조)를 선택하기 위한 초기화어드레스(A11\_RA)를 선택적으로 단위태그테이블(432a ~ 432i)에 출력하는 다수의 제2 멀티플렉서(461\_1c, 461\_2c)를 구비한다.

<53> 이하에서는 도6 및 도7을 참조하여 본 실시예에 따른 메모리 장치의 동작을 살펴본다.

<54> 먼저 도6에 도시된 메모리 장치가 초기화 동작모드에 진입하게 되면, 태그블럭(430)에 구비되는 8+1개의 단위태그테이블(432a ~ 432i)을 동시에 선택하게 된다. 8+1개의 단위태그테이블(432a ~ 432i)의 선택은 초기화 선택신호(IE)가 N+1개의 단위태그테이블(432a ~ 432i)에 대응하여 구비되는 N+1개의 논리합게이트(461\_1a, 461\_1b, ...)에 입력되고, 8+1개의 논리합게이트(461\_1a, 461\_1b, ...)에 논리합게이트의 출력이 8+1개의 단위태그테이블(432a ~ 432i)을 동시에 선택하게 되는 것이다.

<55> 이어서, 다수의 제1 멀티플렉서(461\_1a, 461\_1b, ...)는 초기화 선택신호(IE)에 응답하여 초기화 신호(BN0, BN1, ...)를 N+1개의 단위태그테이블(432a, 432b, ...)로 각각 출력한다.

<56> 도6에 도시된 메모리 장치는 초기화 동작이 끝난 후에는 9개의 단위태그테이블(432a, 432b, ...)중 8개는 각각 제1 ~ 8 논리적 셀블럭어드레스를 저장하고 있고, 하나의 단위태그테이블은 데이터 채저장을 위한 예비용 셀블럭이 된다.

<57> 따라서 초기화 동작시에 9개의 단위태그테이블(432a, 432b, ...)중 8개는 서로 다른 논리적 셀블럭어드레스를 저장하게 된다. 예를 들어 제1 단위태그테이블(432a)에 구비된 256개의 레지스터는 제1 논리적 셀블럭어드레스를 저장하고 있게 되고, 제2 단위태그테이블(432b)는 제2 논리적 셀블럭어드레스를 저장하게 있게 되는 것이다.

<58> 이를 자세히 살펴보면, 전술한 바와 같이 하나의 단위태그테이블은 하나의 단위셀블럭에 구비되는 256개의 워드라인이 각각 어떤 논리적 셀블럭에 대응되는 것인가를 저장하고 있기 때

문에 각각 3비트를 저장할 수 있는 256개의 레지스터를 구비하게 된다. 따라서 초기화 동작시 제1 단위태그테이블(432a)에 구비되는 256개의 레지스터에는 제1 논리적 셀블럭어드레스에 해당되는 '0'이 각각 저장되고, 제2 단위태그테이블(432b)에 구비되는 256개의 레지스터에는 제2 논리적 셀블럭어드레스에 해당되는 '1'이 각각 저장되는 것이다.

<59> 하나의 단위태그테이블에 구비되는 256개의 레지스터에 각각 초기화되는 하나의 논리적 셀블럭어드레스를 저장하기 위해서는 제2 멀티플렉서(461\_1c, 461\_2c, ...)에서 초기화 선택신호(IE)에 응답하여 초기화 어드레스(A11\_RA)가 단위태그테이블로 입력된다.

<60> 단위태그테이블은 초기화 어드레스(A11\_RA)이 입력되면, 구비된 256개의 레지스터에 입력되는 초기화신호(예를 들어 BN0)를 저장하게 된다.

<61> 초기화 동작이 완료된 상태에는 제1 내지 제8 단위태그테이블(432a ~ 432h)의 레지스터에는 0 ~ 7이 각각 저장되어 있게 된다. 9번째 단위태그테이블(432i)에 구비되는 256개의 레지스터는 초기화 동작시 초기화값을 저장하기 않아도 되는데, 이는 9번째 단위태그테이블(432i)에 구비되는 256개의 레지스터는 예비워드라인에 대응하기 때문에 논리적 셀블럭 어드레스를 저장할 필요가 없는 것이다.

<62> 따라서 종래에 단위태그테이블마다 256번을 실시하고 모든 단위태그테이블마다 초기화동작을 따로 실시함으로서 태그블럭의 초기화 동작에 많은 시간이 소모되었던 것에 비하여, 본 발명에서는 태그블럭에 구비되는 모든 단위태그테이블에 한번에 초기화동작을 실시할 수 있으므로 태그블럭의 초기화동작을 크게 단축시켰다.

<63> 이상에서 설명한 본 발명은 전술한 실시예 및 첨부된 도면에 의해 한정되는 것이 아니고, 본 발명의 기술적 사상을 벗어나지 않는 범위 내에서 여러 가지 치환, 변형 및 변경이

가능하다는 것이 본 발명이 속하는 기술분야에서 통상의 지식을 가진 자에게 있어 명백할 것이다.

### 【발명의 효과】

<64> 본 발명에 의해서 태그메모리를 구비하여 고속으로 동작하는 메모리 장치에서 태그메모리의 초기화 동작을 크게 줄일 수 있어 메모리 장치의 동작속도를 더욱 향상시킬 수 있게 되었다.

**【특허청구범위】****【청구항 1】**

각각 M개의 워드라인을 가지며, 입력되는 로우어드레스에 대응하도록 구비된 N개의 단위 셀블럭에, 추가적으로 M개의 워드라인을 가지기 위해 추가의 단위셀블럭을 더 포함하여 N+1개의 단위셀블럭으로 구성되는 셀블럭;

상기 (N+1) 개의 워드라인 중에서, 적어도 어느 한 워드라인을 예비워드라인으로 할 지의 여부에 대한 정보를 저장하기 위한 예비셀블럭 테이블;

상기 로우어드레스를 입력받아 N개의 단위셀블럭을 선택하기 위한 논리적 셀블럭어드레스를 감지하고, 이를 상기 N+1개의 단위셀블럭중 하나를 선택하기 위한 물리적 셀블럭어드레스로 변환하여 출력하는 태그블럭; 및

상기 물리적 셀블럭어드레스에 의해 선택된 단위셀블럭에서의 한 워드라인과, 상기 워드라인에 대한 -상기 예비셀블럭 테이블로부터 제공되는 정보에 의해 정해지는- 예비워드라인을 활성화시키기 위해 상기 태그블럭과 상기 예비셀블럭 테이블을 제어하는 제어수단을 구비하며, 상기 태그블럭은

상기 N+1개의 단위셀블럭에 각각 대응하며, 단위셀블럭에 각각 구비된 256개의 워드라인이 어떤 논리적 셀블럭어드레스에 대응되는지를 저장하기 위한 N+1개의 단위태그테이블; 및 한번의 초기화 동작으로 상기 N+1개의 단위태그테이블을 초기화시키기 위한 초기화 셋팅 수단을 구비하는 메모리 장치.

**【청구항 2】**

제 1 항에 있어서,

### 상기 초기화 셋팅수단은

상기  $N+1$ 개의 단위태그테이블에 대응하며, 일측 입력으로 상기  $N+1$ 개의 단위태그테이블 모두를 선택하기 위한 초기화 선택신호를 입력받고, 타입력으로 상기  $N+1$ 개의 단위태그테이블 중 하나를 선택하기 위한 선택신호를 각각 입력받는 다수의 논리합 수단;

상기  $N+1$ 개의 단위태그테이블에 대응하며, 상기 초기화 선택신호에 응답하여 상기 논리적 셀블럭어드레스와 상기  $N+1$ 개의 단위태그테이블을 초기화시키기 위한 초기화 신호를 선택적으로 단위태그테이블에 출력하는 다수의 제1 멀티플렉서; 및

상기  $N+1$ 개의 단위태그테이블에 대응하여, 상기 초기화 선택신호에 응답하여 상기 단위 셀블럭에 구비되는  $M$ 개의 워드라인을 선택하기 위한 로컬어드레스와, 하나의 단위태그테이블에 구비되는 모든 레지스터를 선택하기 위한 초기화어드레스를 선택적으로 단위태그테이블에 출력하는 다수의 제2 멀티플렉서를 구비하는 것을 특징으로 하는 메모리 장치.

### 【청구항 3】

제 1 항의 메모리 장치를 구동하는 방법에 있어서,

상기 메모리 장치의 초기화 동작모드에 진입하는 제1 단계;

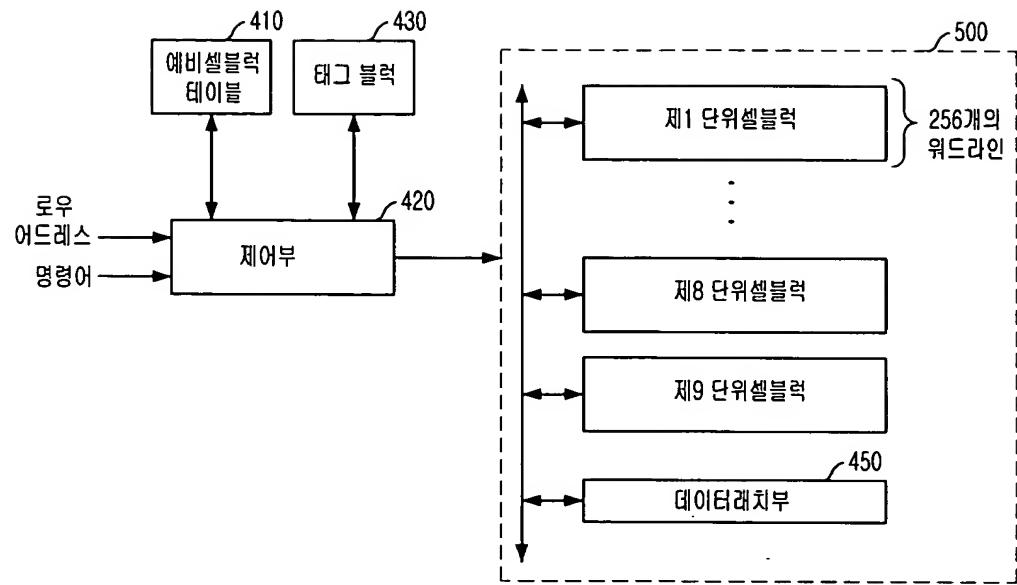
상기  $N+1$ 개의 단위태그테이블을 선택하는 제2 단계; 및

상기  $N+1$ 개의 단위태그테이블 중  $N$ 개의 단위태그테이블에 각각 서로다른 논리적 셀블럭 어드레스를 저장하는 제3 단계

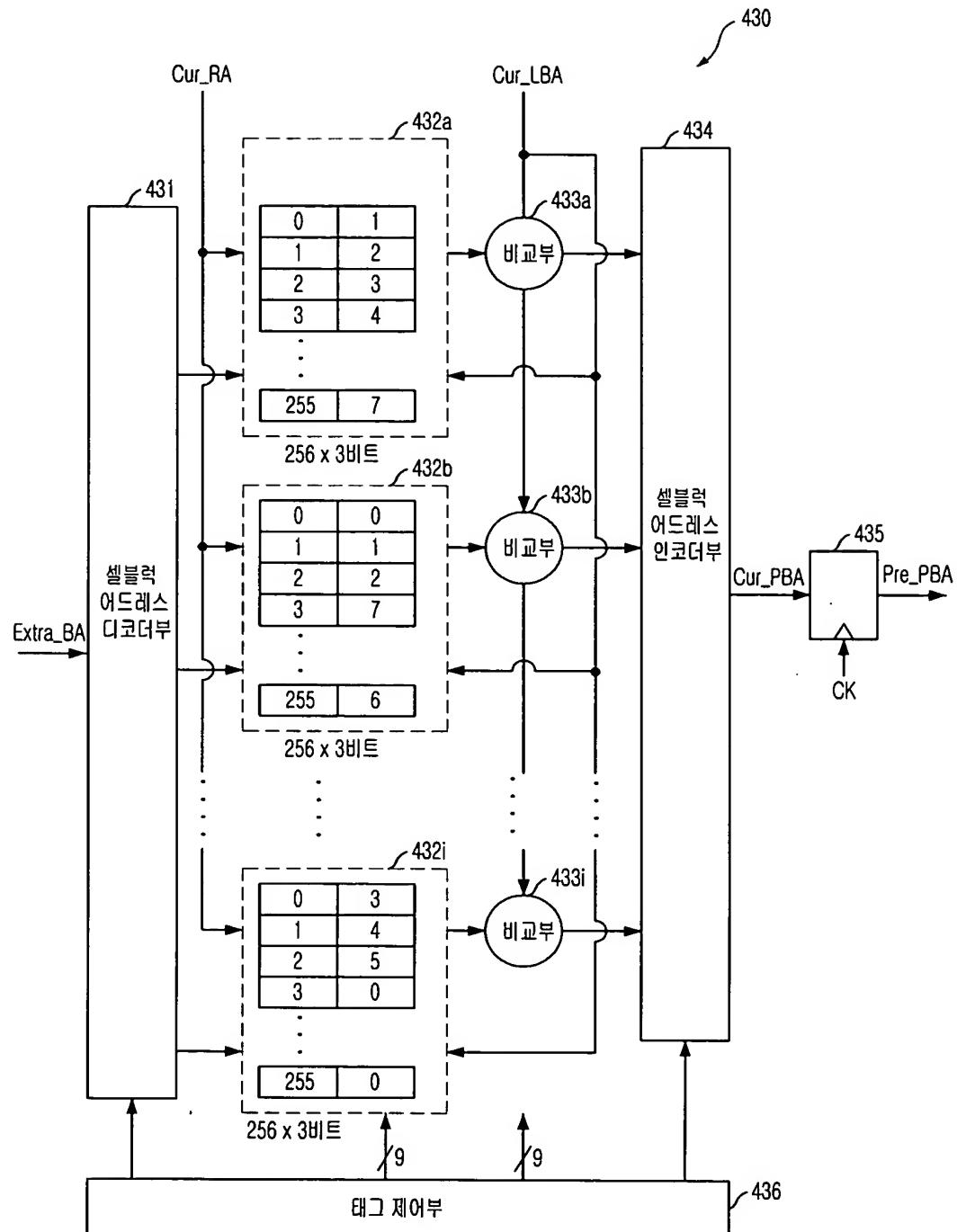
를 포함하는 메모리 장치의 구동방법.

## 【도면】

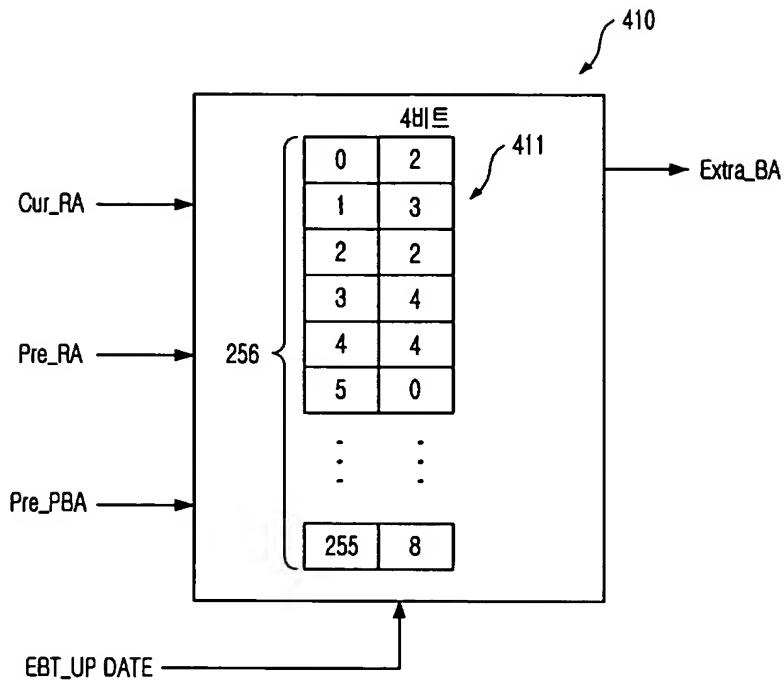
【도 1】



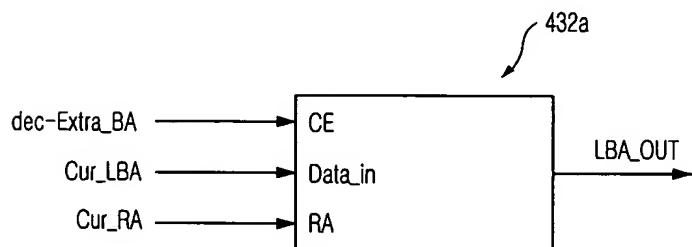
## 【도 2】



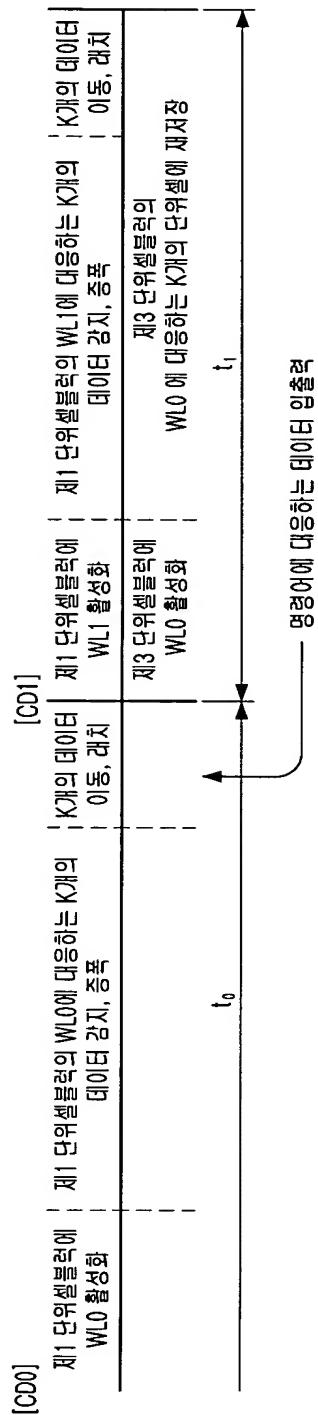
【도 3】



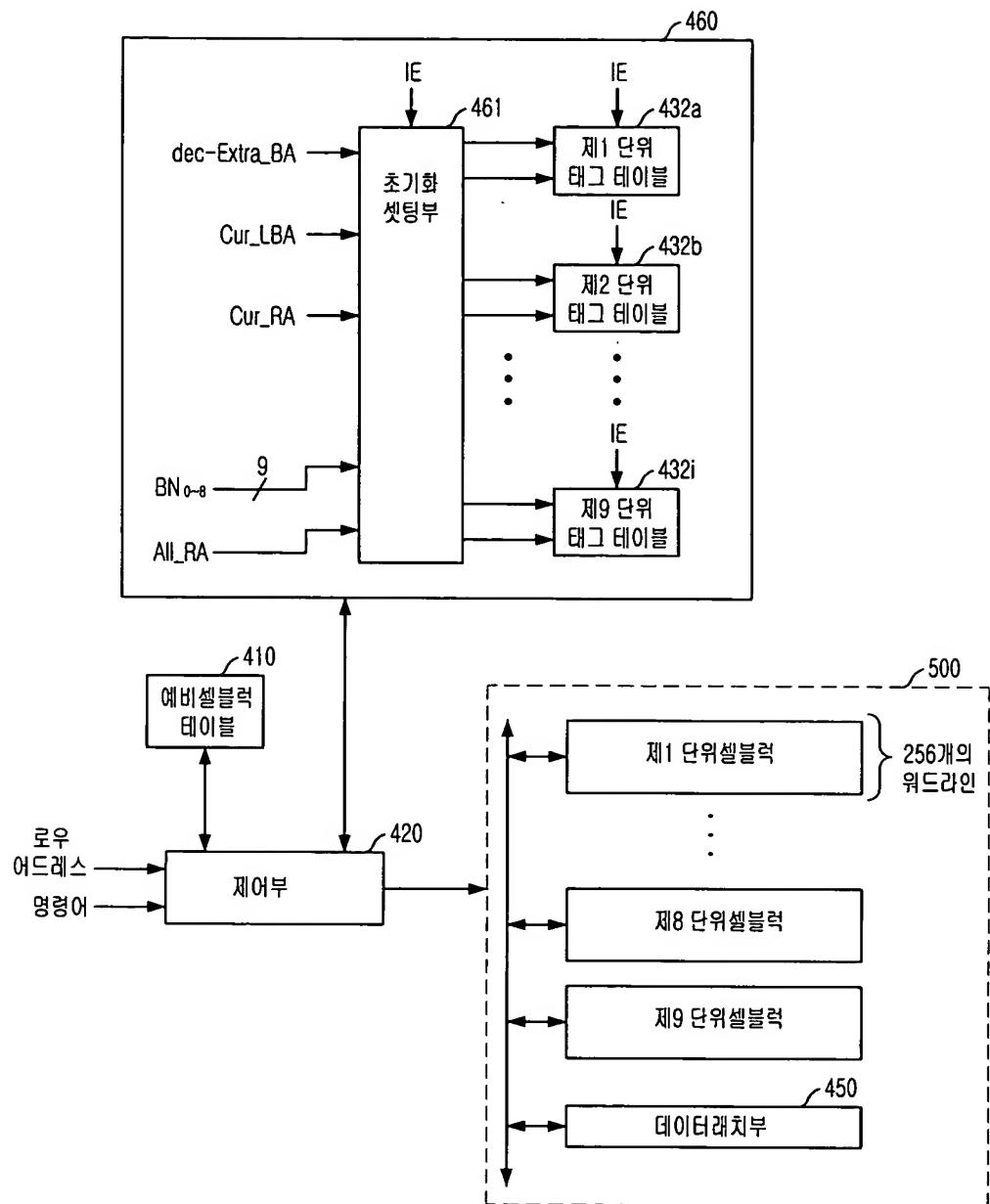
【도 4】



### 【도 5】



## 【도 6】



【도 7】

